ECOLE POLYTECHNIQUE
PROMOTION X2001
BOUREZ Christopher

RAPPORT DE STAGE D'OPTION SCIENTIFIQUE

Etude d'une méthode de segmentation basée sur les contours actifs et la décomposition en ondelettes d'une image.

Détection de vertèbres dans les images radiographiques médicales

NON CONFIDENTIEL

Département de Mathématiques Appliquées
Traitement d'images

Directeur de l´option : Emmanuel Bacry
Directeur de stage : Valérie Perrier

Dates du stage : 10 mai - 31 juillet 2004

Adresse de l´organisme :

Laboratoire de Modélisation et Calculs
Institut d'Informatique et Mathématiques Appliquées de Grenoble
B.P. 53
38041 Grenoble Cedex 9

Résumé L'étude porte sur la méthode des contours actifs, qui depuis l'idée originelle de Kass, Witkin et Terzopoulos, a connu des améliorations considérables, mais au prix parfois d'une augmentation de la complexité. Couplée aux résultats des détecteurs de contours, elle propose une solution au problème de détection de vertèbres sur des images radiographiques médicales et permet de corréler la projection des modèles statistiques de vertèbres avec la radiographie, afin de guider le logiciel de reconstruction 3D.

Abstract The purpose of this work is to study active contour methods, which have been considerably improved since the original idea of Kass, Witkin and Terzopoulos, sometimes to the detriment of a low complexity. Using edge detectors and the multi-scale results, the study leads to a solution to the detection problem of vertebras in radiographic images, and enables to match the 2D-projection of statistic models of vertebras with the radiography, in order to guide the 3D-reconstruction software.

Table des matières

Chapitre 1

Objet de l'étude

Le projet conduit par l'équipe de Valérie Perrier à l'Institut d'Informatique et de Mathématiques Appliquées de Grenoble a pour objectif l'aide au diagnostic médical et la chirurgie assistée par ordinateur. Les résultats de ces recherches intéressent l'industrie et notamment la startup hi-tech Praxim (R) qui développe des stations médicales informatisées pour les blocs opératoires. L'objectif de cette étude est de développer une nouvelle méthode de segmentation de radiographies de vertèbres en mêlant la méthode des contours actifs et la décomposition en ondelettes.

La reconnaissance d'objets, sujet classique en vision par ordinateur [15], s'appuie sur l'extraction de primitives dans les images, tels que des contours ou des régions, ensembles de points aux caractéristiques communes. Le cadre de l'étude se limitera donc aux résultats donnés par les analyses de contours. Des récents travaux de recherche ont été menés au laboratoire sur les différentes méthodes de détection de contours et sont rassemblés dans la thèse d'Olivier Le Cadet. Il en ressort les performances de la transformée en ondelettes mais les données médicales étant bruitées, les contours obtenus sont mal définis ou discontinus.

Le problème posé par l'extraction d'objets à partir des détecteurs de points de contour consiste donc à déduire d'un ensemble de points ou d'une image d'intensités (où l'intensité représente une énergie locale calculée par le détecteur de contours) une courbe fermée continue, délimitant l'objet désiré. Une méthode de recherche des points appartenant au "contour réel", opérée par la minimisation de fonctionnelle, est dite *contours actifs*.

Ainsi, nous commençons par présenter la méthode originelle des contours actifs, son implémentation et ses instabilités (chapitre 1). Des améliorations

comme le flux de vecteurs gradients (GVF) permettent d'augmenter l'efficacité et l'étendue de la méthode (chapitre 2). En amont ou au sein de la minimisation, l'aspect multi-échelle peut être envisagé (chapitre 3). L'étude locale des forces (chapitre 4) donne des résultats nouveaux qui n'existent pas dans la littérature citée en fin (ces développements sont indiqués par des astérisques). Finalement les méthodes développées sont appliquées au problème réel et une solution exploitable s'en dégage pour ce problème complexe.

En annexe, sont données la bibliographie, ainsi que les données relatives au programme : les bibliothèques, le "manuel d'utilisation" et les définitions des principales méthodes implémentées.

NB : pour reconduire les expériences, les programmes Matlab (version 7) et les programmes C sont disponibles sur mon site à l'adresse
`http://christopher.bourez.free.fr/contours_actifs/`.
Pour la compilation, se reporter à l'annexe.

Chapitre 2

La méthode des contours actifs

2.1 Les *snakes*, ou contours actifs

Les premiers modèles de contours actifs ont été développés par Michael Kass, Andrew Witkin et Demetri Terzopoulos [1]. Le *snake* est défini comme une *spline* minimisant une énergie définie sur une image. De façon équivalente, cela revient à dire que son comportement est sous l'influence des forces dérivées de ces énergies, auxquelles on peut rajouter des forces de contraintes extérieures (cela afin de prendre en compte une éventuelle interaction avec l'utilisateur).

La fonctionnelle de l'énergie du contour est décomposée de la façon suivante sur le contour paramétré par $s \in [0, 1]$:

$$E = \int_0^1 E(\vec{u}(s))ds = \int_0^1 (E_{int} + E_{image} + E_{cst})d\vec{u} \tag{2.1}$$

L'énergie interne, qui dépend de la forme prise par le contour, est décomposée en deux termes, une énergie élastique qui tend à rétrécir et régulariser (ie rendre continu) le contour (α définit l'élasticité) et une énergie de courbure qui contrôle la rigidité du contour actif.

$$E_{int} = \frac{1}{2}(\alpha|\vec{u}\,'(s)|^2 + \beta|\vec{u}\,''(s)|^2) \tag{2.2}$$

L'énergie E_{image}, dérivée des données de l'image locales au contour, est classiquement décrite sous la forme

$$E_{image} = w_{line}E_{line} + w_{edge}E_{edge} + w_{term}E_{term}$$

où

$$\begin{aligned} E_{line} &= I(x,y) \\ E_{edge} &= -|\overrightarrow{grad}I(x,y)|^2 \\ E_{term} &= \frac{\partial\theta}{\partial\vec{n}_R} \end{aligned}$$

La fonctionnelle (*line*) pousse le contour vers les lignes foncées ou claires en fonction du signe de w_{line}, (*edge*) vers les forts contours (lignes de forts gradients) et la dernière vers les terminaisons et coins (où le rayon de courbure est infini). Enfin, l'énergie E_{cst} permet d'introduire des points d'attractions élastiques (en $-kx^2$) ou des puits de potentiels (en $-k/r^2$).

Portée de l'énergie, du champ d'attraction associé Si l'image est globalement homogène en dehors de quelques zones de discontinuités correspondant à des contours, on voit bien que E_{line} est constant, E_{edge} nul et E_{term} nul (car indéfini). Dans ce cas les données de l'image n'agiront pas, par l'intermédiaire de la minimisation, sur la forme du contour actif. Pour s'assurer que le contour soit correctement attiré vers les lignes, les contours et les terminaisons, il faut dépasser le problème de minimisation et confronter recherche locale / recherche globale.

Plus clairement, une solution consiste à rendre l'image floue afin de diffuser les contours. On peut pour cela lisser les fonctionnelles E par un filtre gaussien de déviation standard σ, et éventuellement réduire l'échelle au cours de la minimisation

$$E^*_{line} = G_\sigma \star E_{line}$$

$$E^*_{edge} = -\overrightarrow{\nabla}(G_\sigma \star I(x,y))$$

Dans le chapitre 3, on introduira une méthode récente, le flux de vecteurs gradients.

Remarquons que l'échantillonnage de l'image n'est parfois pas nécessaire si les images sont suffisamment dégradées par la technologie et la digitalisation [16]. Dans le cas contraire le filtre permettra d'éliminer le bruit, qui pourrait fortement gêner l'évolution du contour actif. En modélisant le flou par un filtre gaussien, il est possible, à partir d'une image, de déduire la variance du flou [17].

Lien entre les filtres gaussiens et les ondelettes Les filtres gaussiens sont très souvent utilisés en traitement d'image, cela permet de régulariser les données (cf. l'action sur une portion de radiographie figure 2.1). On convole avec $G_\sigma = (1/S)exp(-\|x\|^2/\sigma^2)$.

Pour calculer le gradient ∇I de l'intensité de l'image, on peut également appliquer le filtre de Sobel qui moyenne les deux composantes dérivées sur 9 pixels

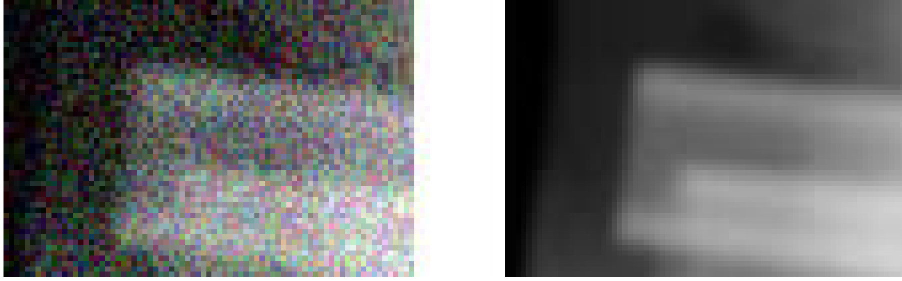


FIG. 2.1 – Une image réelle (bruitée) et l'image traitée par un filtre gaussien adéquat

voisins (que l'on peut également voir comme une "approximation" de la convolution de l'opérateur discret ∇ par G_σ).

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \ , \ S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Les propriétés de convolution donnent

$$E^*_{edge} = -I \star \overrightarrow{\nabla} G_\sigma$$

qui correspond à une transformée en ondelette gaussienne. Ainsi, la théorie des ondelettes va nous apporter un cadre plus général à la résolution du problème.

Le rôle de l'initialisation La difficulté à la minimisation du problème variationnel (2.1) est sa complexité. L'espace de recherche est un espace discret de fonctions, définies sur un ensemble de cardinal k le nombre de points fixé pour le contour actif, à valeur dans l'espace des pixels, soit $N \times M$ positions si l'image est de taille $N \times M$. Une recherche opérationnelle est un problème complexe d'ordre $O((N \times M)^k)$.

Il faut donc avoir une idée du contour initial. A partir de ce contour, si l'on s'autorise de déplacer chaque point du contour de l pixels, la complexité de la recherche opérationnelle tombe à $O(l^k)$.

L'initialisation requiert l'intervention de l'utilisateur. On intégrera la possibilité de faciliter cette intervention, en offrant un panel de formes prédéfinies et un ajustement par translation/rotation/homothétie. Cela permettra un interfaçage avec les travaux d'une thèse à l'IMAG portant sur la création d'une base de données statistiques de modèles de vertèbres, et son logiciel de projection $3D \to 2D$.

2.2 Méthodes numériques & implémentation

Minimisation d'énergie La minimisation de l'énergie (2.1) satisfait l'équation d'Euler

$$\alpha\vec{u}''(s) - \beta\vec{u}''''(s) = \overrightarrow{\nabla}(E_{image} + E_{cst}) = \overrightarrow{\nabla}E_{ext} \tag{2.3}$$

En effet, le problème (2.1) revient à minimiser $J(u) = \frac{1}{2}a(u,u) + f(u)$, pour $u \in K = (H^1([0,1]))^2$ avec

$$a(u,v) = \int_0^1 (\alpha u'(s) \cdot v'(s) + \beta u''(s) \cdot v''(s))ds$$

et

$$f(u) = \int_0^1 E_{ext}(u(s))ds$$

Si J est continue et différentiable, le minimum local se traduit par la condition $J'(u)(v-u) \geq 0$ pour tout v dans l'espace de Hilbert K, soit si $u \in K \setminus \partial K$ $J'(u) = 0$. Soit

$$\forall h \int_0^1 (\alpha u'(s) \cdot h'(s) + \beta u''(s) \cdot h''(s))ds = -\int^1 \nabla E_{ext}(u(s))h(s)ds$$

On introduit, comme expliqué dans [14], l'espace $H_0^1 = \{h \in H^1/h(0) = h(1) = 0, h'(0) = h'(1) = 0\}$. En appliquant la formule de Green (intégration par parties) on obtient l'égalité suivante au sens des distributions

$$(\alpha u')' - (\beta u'')'' = \nabla E_{ext}$$

dont la forme pour α et β constants est (2.3).

La théorie de Lax-Milgram [14] propose un cadre théorique à la démonstration de l'unicité du minimum global sous certaines conditions. En pratique l'espace K est un espace de fonctions dont les espaces de départ et d'arrivée sont discrets et bornés, il est donc fini. Il existe donc bien un minimum global au problème mais il n'a pas d'intérêt pratique. Ce minimum n'est pas toujours unique, dans le cas d'une fonction f constante [ou linéaire], l'ensemble des minima contient toutes les fonctions constantes de K [resp. la fonction nulle], correspondant à des contours nuls. En fait, toute la difficulté sera de trouver le minimum local adéquat parmi des minima locaux admissibles et de trouver un critère d'admissibilité.

Iterations On introduit un paramètre t pour obtenir une résolution itérative à partir d'une solution initiale (donnée en phase d'*initialisation*),

$$\frac{\partial \vec{u}}{\partial t} = \alpha\nabla^2\vec{u}(s) - \beta\nabla^4\vec{u}(s) - \overrightarrow{\nabla}E_{ext}$$

en espérant une convergence en temps de l'algorithme. Dans le cas $\beta = 0$ chaque composante u de $\vec{u}$ vérifie l'équation

$$\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial s^2} = F(u) \tag{2.4}$$

Dans le cadre du continu on évite l'instabilité et l'imprécision due à la diffusion dans les différences finies, en utilisant l'analyse spectrale comme dans la résolution des équations de la chaleur

$$\frac{d\hat{u}_K(t)}{dt} + \alpha' \hat{u}_K(t) = \hat{F}(u)_K$$

avec $\alpha' = -\frac{4\pi^2 K^2}{N^2 h^2}\alpha$. Comme dans les équations de Navier-Stokes, on résoud l'équation précédente entre t-dt et t+dt.

$$\hat{u}_K(t+dt) = \hat{u}_K(t-dt)exp(-2\alpha' dt) + \int_{t-dt}^{t+dt} exp(-\alpha'(t+dt-s))\hat{F}(u)_K(s)ds$$

L'approximation des rectangles conduit à

$$\hat{u}_K(t+dt) = \hat{u}_K(t-dt)exp(-2\alpha' dt) + 2dt\hat{F}(u)_K(t)exp(-\alpha' dt) \tag{2.5}$$

Les différences finies Classiquement on choisit un schéma d'Euler implicite pour le terme de diffusion dans l'équation (2.4), évitant la condition CFL [14] ; pour conserver un schéma simple et inversible dans le cas général, on a pris un schéma explicite pour le dernier terme non linéaire ∇E_{ext} qui dépend de l'image. On obtient la discrétisation d'ordre 2 suivante

$$\frac{u_i^t - u_i^{t-1}}{\tau} = \alpha \frac{u_{i+1}^t - 2u_i^t + u_{i-1}^t}{h^2} + F(\vec{u}_i^{\,t-1}) \tag{2.6}$$

pour chaque composante u de $\vec{u}$.

Décomposition LU et inversion Le résultat discrétisé (2.6) se retrouve dans l'article original [1] exprimé sous la forme

$$(I + \tau A)U^t = U^{t-1} + \tau F(\vec{U}^{t-1})$$

où $U = (x(i))_i$ ou $(y(i))_i$ le vecteur des abscisses ou des ordonnées de $\vec{u}$. A est une matrice pentadiagonale avec trois termes supplémentaires dans chaque coin afin de prendre en compte la périodicité.

La première solution pour la résolution de ce système, dont une implémentation est proposée dans le programme C, est l'inversion LU de la matrice $M = I + \tau A$ du système. On retrouve facilement la formule générale pour un choix $L_{i,i} = 1$

$$L_{i,j} = U_{j,j}^{-1}(M_{i,j} - \sum_{k<j} L_{i,j}U_{k,j}) \ \ j \leq i-1$$

$$U_{i,j} = M_{i,j} - \sum_{k<i} L_{i,k}U_{k,j} \ \ j \geq i$$

On peut effectuer un *pivoting* qui consiste à réorganiser les lignes (la solution finale restera inchangée) afin de placer les termes les plus grands sur la diagonale et d'éviter des erreurs numériques (dues à un terme $U_{j,j}^{-1}$ trop grand).

Le calcul de $U_{i,j}$ compte i additions et $(i-1)$ multiplications, effectuées pour les $n+1-i$ termes de la ligne, soit $(2i-1)(n+1-i)$ opérations. En posant $f(x) = \sum_{k\leq n} x^k = (1-x^{n+1})/(1-x)$ on voit que $(x^{n-1}f'(1/x))' = \sum_k k(n-k)x^{n-k-1}$ et donc à la limite lorsque $x \to 1$, $\sum_{k\leq n} k(n-k) = 1/6(n^3-n)$. La complexité du calcul de L étant la même que celle de U, on obtient une complexité pour l'algorithme LU de l'ordre de $O(N^3)$ où N est le nombre de points du contour. La nullité de nombreuses valeurs de M n'a d'ailleurs pas d'impact positif sur la forte complexité de l'inversion par cette méthode. Remarquons néanmoins, qu'à paramètres fixés (ie l'élasticité, le nombre de points du contour), la matrice peut être calculée une fois pour toute, au moment du lancement du programme, "indépendamment" des images à traiter.

Implémentation FFT (Matlab) On peut remédier à la complexité de l'inversion par l'implémentation d'une FFT.

L'écriture spectrale de la discrétisation (2.6) est

$$\frac{\hat{u}_s^t - \hat{u}_s^{t-1}}{\tau} = \alpha\frac{\hat{u}_{s-1}^t + \hat{u}_{s+1}^t - 2\hat{u}_s^t}{h^2} + \widehat{F(u^{t-1})}$$

d'où, pour un contour de longueur N, comme $\hat{u}_{s+1}^t(k) = e^{(2i\pi k/N)}\hat{u}_s^t(k)$

$$\left(1 + 2\alpha\tau/h^2 sin^2(\frac{\pi k}{N})\right)\hat{u}^t = \tau\widehat{F(u^{t-1})} + \hat{u}^{t-1}$$

La complexité est alors donnée par la transformée de Fourrier discrète, soit $O(NlogN)$.

Lien entre Fourrier continu et Fourrier DF Lorsque $N \to +\infty$, la linéarisation du coefficient de diffusion $2\alpha/h^2 sin^2(\pi k/N)$ dans cette dernière équation due aux différences finies, redonne le coefficient de diffusion en continu (expression 2.5), qui était $(exp(-\alpha'2dt)-1)/2dt \sim -\alpha' = \frac{4\pi^2K^2}{N^2h^2}\alpha$.

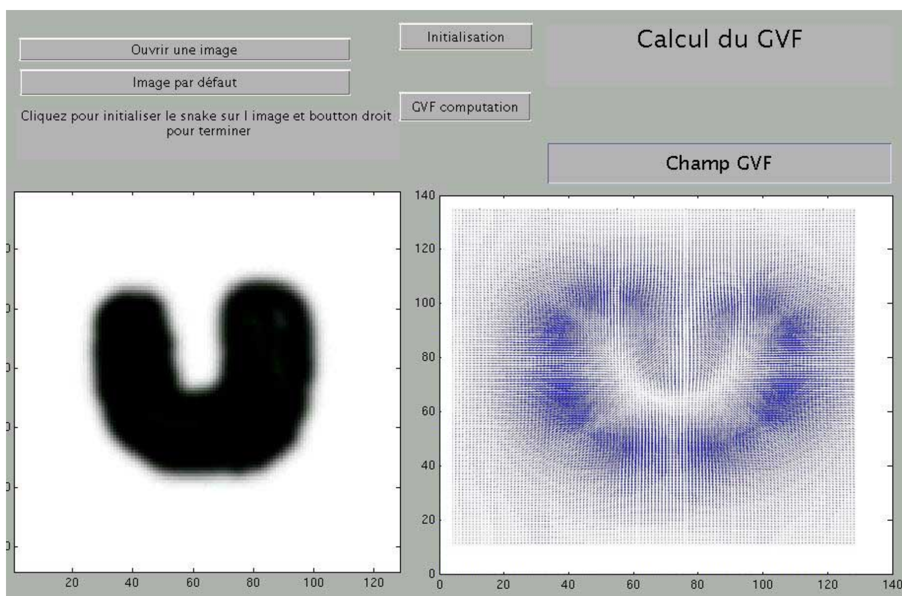


FIG. 2.2 – Interface Matlab

2.2.1 Interprétation

Bilan des forces L'équation (2.3) s'interprète classiquement sous la forme

$$\vec{F}_{int} + \vec{F}_{ext} = 0$$

où $\vec{F}_{ext} = \vec{\nabla} E_{ext}$, il s'agit de l'équilibre résultat de l'évolution régie par l'équation

$$\frac{\partial u}{\partial t} = \vec{F}_{int} + \vec{F}_{ext}$$

En quelque sorte, le contour actif évolue tel une ligne matérielle sous l'action d'un champ de force $\vec{F}_{ext}$ et de contraintes internes $\vec{F}_{int}$. Les termes d'élasticité/rigidité α et β imposent une sorte de "continuité" ; plus précisément les régularités respectives introduites sont des dérivabilités d'ordre 2 et 4.

D'autres formes d'énergies internes ont été développées à des fins particulières, par exemple pour la création de "bras" dans le but de la détection de routes [13]. Ces énergies, calculées par l'intégrale curviligne de quadratiques sont néanmoins complexes à étudier, ne permettant pas l'extraction d'une équation d'évolution comme (2.4) et nécessitant des algorithmes de recherche opérationnelle.

Plus simplement, on peut remarquer qu'il est possible de jouer sur la forme du contour, en adaptant les fonctions α et β localement. En un point d'annulation de α et β, on autorise une discontinuité, un "coin". Avec un nombre fini de points

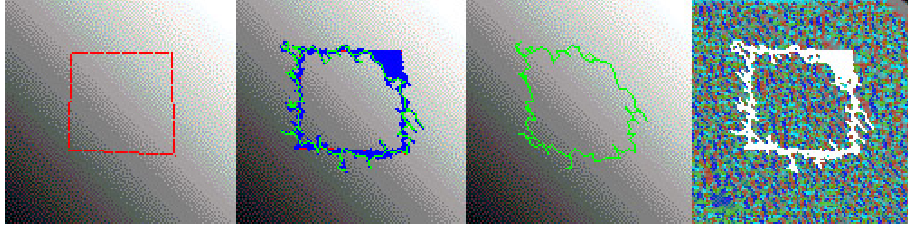


FIG. 2.3 – Simulation d'instabilité.
De gauche à droite : intialisation, évolution, final, trajet de l'évolution sur fond de champ d'attraction
[convention de représentation du champ d'attraction : l'intensité lumineuse représente l'intensité du champ, la couleur la direction du champ]

d'annulation de α et β on peut donc décrire facilement des polygones. Mais encore faut-il que cela ait une signification, c'est-à-dire qu'il y ait une raison pour que le coin du polygone soit correctement attiré par un coin du contour sur l'image.

Instabilité numérique La méthode à l'état brut est très instable (figure 2.2). Cela vient de la faible portée du champ d'attraction vers les contours, ou, en inversant le problème, d'une initialisation du contour actif pas suffisamment précise. Dans les zones homogènes, le contour actif va privilégier la minimisation de son énergie interne et s'effondrer sur lui-même : en effet, à E_{ext} constante, le contour converge vers un point (en tendant vers une forme circulaire).

Autre limite, les formes concaves Une autre formulation du problème introduit par un terme d'élasticité trop fort est l'absence de convergence dans les concavités (figure 2.3). Le champ est faible dans la concavité et le contour actif privilégie à nouveau la minimisation de l'énergie interne.

Or, malgré les deux problèmes mentionnés, on ne peut réduire l'élasticité α du contour. Même l'idée de réduire l'élasticité lorsque l'initialisation est proche des contours réels conduit à des mauvais résultats, avec un contour fortement discontinu, un escalier qui suit localement le contour des pixels et diverge en serpentant énormément, suivant les discontinuités numériques et notamment le bruit. L'élasticité a donc pour but d'introduire une certaine contrainte de contour afin que la forme du contour ne dépende pas du bruit.

Désormais, nous nous concentrerons sur l'amélioration du champ d'attraction $\vec{F}_{ext}$.

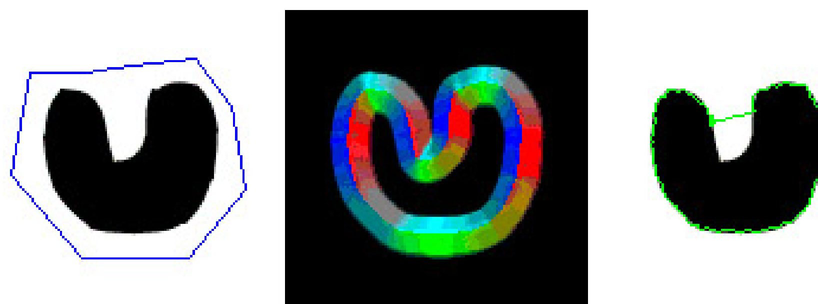


FIG. 2.4 – Les concavités (zones de champ faible) engendrent des difficultés.
De gauche à droite : initialisation, champ d'attraction, contour final
On remarque également que l'attraction a peu de portée.

Chapitre 3

Diffusion d'une information localisée

Les difficultés rencontrées avec la méthode classique des contours actifs viennent d'un manque d'indications dans les zones homogènes et lisses qui traverserait le contour actif. Chenyang Xu et JL Prince [8][9][10] proposent une méthode innovante pour élargir le champ d'attraction de l'énergie potentielle. Nous comparerons cette méthode avec les traditionnels lissages gaussiens.

3.1 Une formulation du problème*

Le contrôle du comportement du contour peut être divisé en deux problèmes

Les points d'attraction Il s'agit de définir un sous-ensemble S de pixels présentant une caractéristique d'appartenance à des contours (par exemple les forts gradients). Basé sur le principe d'Helmholtz, J-M Morel [15] propose des calculs statistiques propres aux fonctionnement de l'oeil humain dans le cadre de la détection d'alignements significatifs. Pour des formes quelconques, il est encore difficile d'adopter ces calculs statistiques. La méthode des contours actifs est souvent associée aux E_{int} du premier chapitre. Le chapitre suivant étudiera le problème plus en profondeur, notamment avec l'introduction des détecteurs multiéchelles.

La "ligne" Le contour actif sert d'outil pour extraire des lignes fermées. Mais l'information du contour est une information locale (voire ponctuelle) et n'a pas de "portée d'attraction". Elle doit donc être diffusée. Par conséquent, étant donné une intensité ou une énergie I, l'objectif est de déduire de I une fonction P et sous-ensemble S' de S le plus grand possible, vérifiant au

mieux la propriété

$$\forall x \in S' \quad P(x) = I(x) \;\; \nabla P(x) = 0 \;\; \Delta P(x) < 0$$

3.2 Les filtres gaussiens

C'est la solution la plus simple au problème de la "diffusion" du contour. Ils présentent un inconvénient majeur : la "diffusion" du contour s'effectue avec perte d'information et il n'est pas possible de dissocier les deux. En effet, si l'on applique un filtre gaussien 1D $exp(-x^2/\sigma^2)$ sur la fonction $\delta + h\delta_t$ (où δ_t est le Dirac centré en t), on vérifie facilement que les deux Diracs sont confondus en un seul pic à partir d'une distance t inférieure à t_{min} solution de

$$h\, e^{\frac{t_{min}^2 - 1}{\sigma^2}} = 2\frac{t_{min}^2}{\sigma^2} - 1$$

pour $h < 1$.

Pour une diffusion de variance σ, on introduit une imprécision liée à σ. Cela est illustré dans la figure 3.1.

Remarque Dans ce cas, on peut justifier le terme littéraire de "diffusion" de façon mathématique : la solution de l'équation de diffusion de la chaleur est à vrai dire une convolution de la condition initiale par une gaussienne [17].

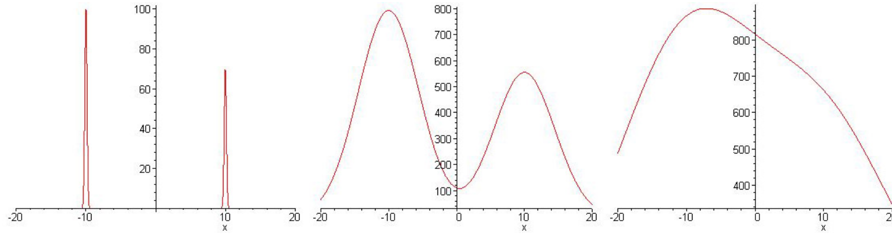


Fig. 3.1 – Deux filtres gaussiens appliqués sur un signal 1D. La "diffusion" de l'information (par suppression des hautes fréquences) entraine une perte d'information sur les discontinuités.

3.3 Gradient Vector Flow

3.3.1 Intuition

Pour gagner en performances, Xu and Prince ont eu l'idée de remplacer le champ ∇E de rotationnel nul par un champ plus général. A partir d'une carte de contours f comme ∇E, ils ont défini un nouveau champ $\mathbf{v}$ = (u,v) comme la solution de la minimisation de la fonctionnelle

$$\int\int \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |\mathbf{v} - \nabla f|^2 \tag{3.1}$$

afin que $\mathbf{v}$ ait une faible variation là où $|\nabla f|$ est faible. La conséquence est la diffusion des forts gradients vers les zones homogènes (gradient faible).

L'équation d'évolution Le résultat (3.4) de la section suivante dans le cas général montre que chaque composante v_x et v_y de $\mathbf{v}$ est solution de l'équation suivante, similaire à (2.4) :

$$\frac{\partial u}{\partial t} - \mu \Delta u = F_{x/y}(u) \tag{3.2}$$

avec $F_x(u) = -(u - f_x)|\nabla f|^2$.

Implementation FFT (Matlab) On discrétise l'équation obtenue (3.2) par les différences finies comme en (2.6) :

$$\frac{\hat{u}_{i,j}^t - \hat{u}_{i,j}^{t-1}}{\tau} = \mu(\hat{u}_{i-1,j}^t + \hat{u}_{i+1,j}^t + \hat{u}_{i,j-1}^t + \hat{u}_{i,j-1}^t - 4\hat{u}_{i,j}^t) + \widehat{F(u^{t-1})}$$

et sachant que $\hat{u}_{i+1,j+1}^t(k,l) = e^{(2i\pi k/N)} e^{(2i\pi l/M)} \hat{u}_{i,j}^t(k,l)$ pour une image de taille $N \times M$

$$\left(2\mu\tau \left(sin^2(\frac{\pi k}{N}) + sin^2(\frac{\pi l}{M})\right) - 1\right) \hat{u}^t = -\tau \widehat{F(u^{t-1})} - \hat{u}^{t-1}$$

3.3.2 GVF généralisé

La forme généralisée de (3.1) est

$$\int\int_{image} g(|\nabla f|)(u_x^2 + u_y^2 + v_x^2 + v_y^2) + h(|\nabla f|)|\mathbf{v} - \nabla f|^2 \quad dxdy \tag{3.3}$$

Ainsi, à la place d'un paramètre μ fixe, on peut choisir l'importance de la "régularité" de la solution par l'intermédiaire de f et h. Il faut garder à l'esprit

d'accentuer la diffusion dans les zones homogènes (où $|\nabla f|$ est faible) en minimisant le premier terme $(u_x^2+u_y^2+v_x^2+v_y^2)$ de la fonctionnelle et privilégier la minimisation de $|\mathbf{v}-\nabla f|^2$ dans les zones de forts gradients. L'article [8] propose

$$g(|\nabla f|) = exp(-|\nabla f|/K)$$

$$h(|\nabla f|) = 1 - g(|\nabla f|)$$

le paramètre K permettant de contrôler la diffusion.

Du variationnel à l'équation Si l'on note $J(\mathbf{v})$ la fonctionnelle (3.3), que l'on suppose continue et différentiable, la solution $\mathbf{v}$ de la minimisation vérifie $\nabla J(\mathbf{v}) = 0$, où ∇J désigne la différentielle. L'article [11] expose le calcul direct (et coûteux) de cette différentielle mais pour ne pas être redondant, nous proposons ici la méthode classique, en décomposant J sous la forme $J(\mathbf{v}) = \frac{1}{2}a(u,u) + L_x(u) + \frac{1}{2}a(v,v) + L_y(v) + constante$ où

$$a(u,v) = 2\int g\nabla u \cdot \nabla v + huv$$

$$L_x(u) = -2\int huf_x$$

$$L_y(v) = -2\int hvf_y$$

Il ne reste plus qu'à appliquer le raisonnement qui nous avait conduit à l'équation d'Euler 2.3, en appliquant la formule de Green à $\nabla_u J(u)(\phi) = a(u,\phi) + L(\phi)$ pour $\phi \in H^1([image])$, puis par un argument de densité, on obtient $-\nabla \cdot (g\nabla u) + hu = hf_x$ et $\frac{\partial u}{\partial n}_{|bords} = 0$ d'où pour chaque composante

$$\nabla \cdot (g\nabla u) = h(u - f_x) \qquad \frac{\partial u}{\partial n}_{|bords} = 0$$
$$\nabla \cdot (g\nabla v) = h(v - f_y) \qquad \frac{\partial v}{\partial n}_{|bords} = 0$$

Précisons que ces calculs sont valables pour un champ $\mathbf{v}$ recherché dans l'espace produit des espaces de Sobolev d'ordre 2, $(H^2([image]))^2$. On a également supposé que notre espace *image* a été "rogné" dans les coins, afin d'être assimilable à un ouvert de classe C^2.

Stricte convexité On remarque que

$$\begin{aligned}\left(\frac{1}{2}a(u+\epsilon,v+\epsilon)+L_x(u+\epsilon)\right)-\left(\frac{1}{2}a(u,u)+L_x(u)\right) &= \frac{1}{2}a(\epsilon,\epsilon)+a(u,\epsilon)+L_x(\epsilon)\\ &= \frac{1}{2}a(\epsilon,\epsilon)+\nabla_u J(u)(\epsilon)\end{aligned}$$

La même chose est valable pour la composante v. En sommant on obtient

$$J(\mathbf{v}+(\epsilon_1,\epsilon_2)) - J(\mathbf{v}) - \nabla J(\mathbf{v})(\epsilon_1,\epsilon_2) = \frac{1}{2}(a(\epsilon_1,\epsilon_1)+a(\epsilon_2,\epsilon_2)) \geq 0$$

Un autre calcul (en développant les intégrales) est donné dans [11], avec les variations de Gâteaux.

Sous des conditions peu contraignantes, comme par exemple g et h non tous deux nuls en un même point, la forme bilinéaire est définie positive, et la stricte convexité en découle.

Cela implique d'ailleurs l'unicité du champ **v** solution. Remarquons que la forme bilinéaire n'est a priori pas coercive (minorer h et g ne suffirait pas à cause de l'espace de travail H^2 et sa norme compliquée) et on ne peut pas déduire le résultat d'unicité de la théorie de Lax-Milgram.

Ce qui nous intéresse plutôt ici, c'est que s'il y a convergence vers un minimum local, c'est le minimum global.

Implementation C Même si le schéma implicite dû aux différences finies serait similaire à celui du contour actif (2.6), la taille des matrices diffère, il ne s'agit plus du nombre de points du contour mais de la taille de l'image ! Or il est peu recommandable d'inverser un système de la taille de l'image.

Il est plus judicieux de privilégier la vitesse d'exécution et de discrétiser le système suivant un schéma entièrement explicite. Les équations s'écrivaient en tout point (x,y)

$$u_t = g(|\nabla f|)\Delta u - bu - c_1$$
$$v_t = g(|\nabla f|)\Delta v - bv - c_1$$

avec $b = h(|\nabla f|)$, $c_1 = bf_x$ et $c_2 = bf_y$, et la discrétisation de la première (pas de temps τ, pas d'espace h) conduit à

$$u_{i,j}^{n+1} = (1-b_{i,j}\tau)u_{i,j}^n + r_{i,j}(u_{i+1,j}^n + u_{i-1,j}^n + u_{i,j+1}^n + u_{i,j-1}^n - 4u_{i,j}^n) + c_{i,j}^1$$

où

$$r_{i,j} = \frac{g(|\nabla f_{i,j}|)\tau}{h^2}$$

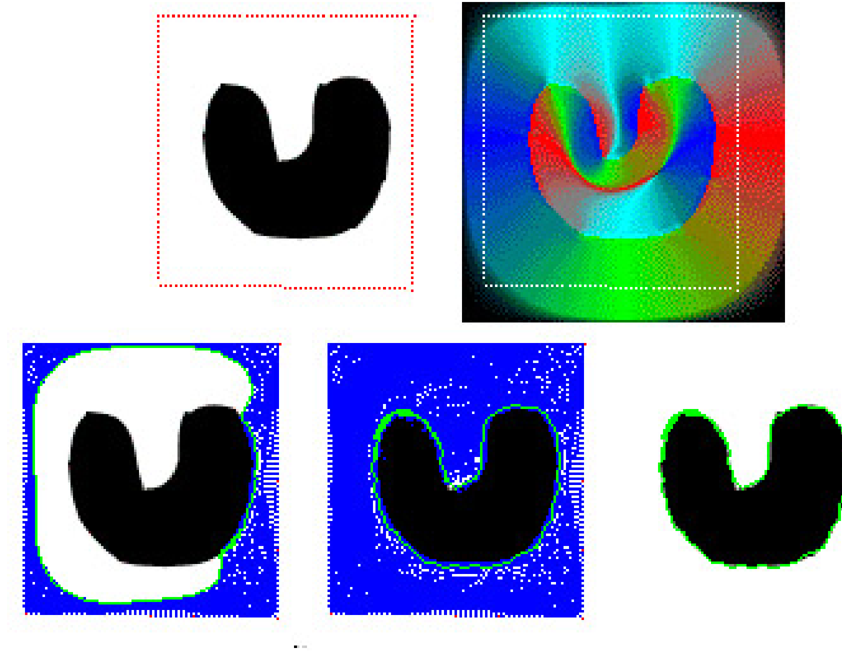


FIG. 3.2 – Portée et attraction dans les concavités grâce au champ GVF
Le contour actif classique n'aurait ni bougé de si loin, ni atteint la concavité
Initialisation (sur fond normal et sur fond de champ d'attraction GVF), évolution, contour final

La stabilité requiert la satisfaction de la condition de Courant-Friedrichs-Lewy $r_{max} \leq 1/4$ soit

$$\tau \leq \frac{h^2}{4g_{max}}$$

3.3.3 Evaluation

Les améliorations La méthode, en conservant les forts gradients et en diffusant dans les zones homogènes, permet d'améliorer significativement les performances du contour actif, et cela contribue à améliorer deux cas de figures

- les concavités
- une initialisation lointaine par rapport aux contours

Ces exemples sont présentés dans la figure (3.2) (ils sont également proposés par les auteurs du GVF [8] à l'adresse `http://iacl.ece.jhu.edu/projects/gvf/`).

De plus, le cas A de la figure 3.4 confirme que deux contours bien marqués sont conservés quelles que soient leur distance et la force de diffusion.

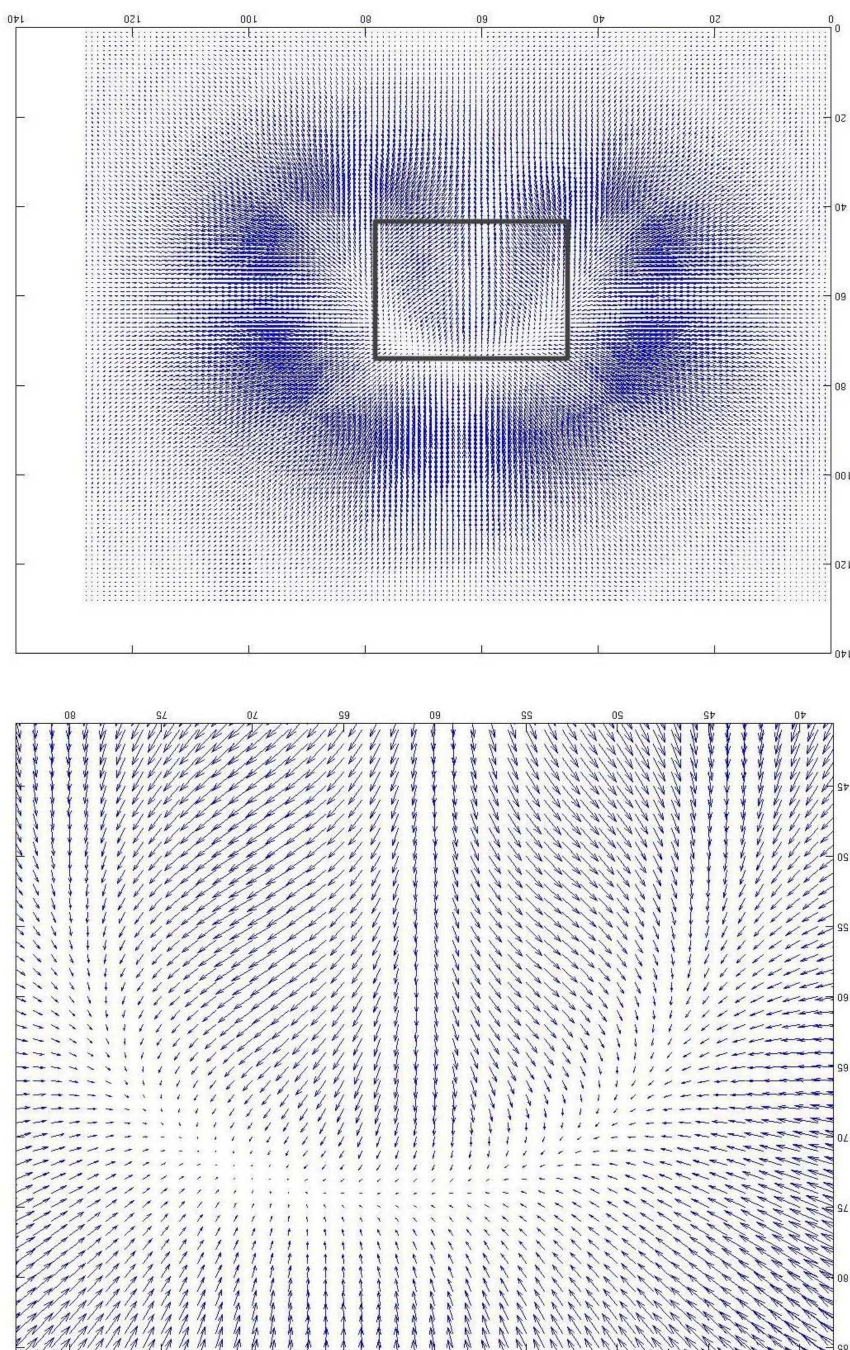


Fig. 3.3 – Lignes de champ du champ d'attraction dans les concavités

Limitations de la diffusion* Le cas B de la figure 3.4 illustre une perte de détails due à la diffusion : les contours les plus faibles s'effacent à faible distance d'un contour bien marqué.

Si l'on veut conserver une convergence précise, la solution (cf. 4.3) est de basculer sur l'image originale lorsque la progression du contour actif s'est ralentie (ie lorsqu'on est plus proche d'un contour).

Limitations - Temps d'exécution* La complexité du calcul à chaque itération est de $O(N \times M)$ si l'image est de taille $N \times M$. La diffusion s'opère de proche en proche, et le développement par différences finies étant de l'ordre 2, la diffusion du champ d'attraction s'effectue sur un carré de 3×3 pixels à chaque itération. Pour avoir une bonne diffusion (sur une distance de la largeur de l'image à un facteur multiplicatif près), l'algorithme s'exécute en au moins $O(N)$ itérations, soit une complexité finale d'au moins $O(N^3)$.

Pour fixer les idées, voici quelques ordres de grandeur : sur un processeur Pentium M 1,70 GHz, le temps d'exécution de 1000 itérations GVF est de moins de 2 secondes pour une image de 128×128, mais pour une image de 1024×1024, les convolutions non optimisées durent une minute, la diffusion suivant chaque axe une minute également.

La diffusion est donc très lente pour des images de grosse taille et il reste à faire le compromis entre portée de l'attraction et rapidité de l'exécution, entre tolérance à l'initialisation et recherche de temps réel. On ne peut donc pas utiliser le GVF pour faire du suivi d'objets ("tracking") dans des séquences video où la méthode des contours actifs était déjà utilisée [12]. Le GVF est à privilégier pour des situations plus statiques.

3.4 Les autres solutions ?*

La littérature donne d'autres exemples de forces f, comme la distance la plus proche aux forts gradients. Cela nécessite l'introduction d'un seuil (pour lutter contre la présence de bruit) pour le choix des forts gradients et il faudra certainement pondérer la distance par l'intensité du gradient.

Il est possible également d'envisager des convolutions plus générales, avec des fonctions non intégrables comme $x \to 1/(1+|x|)$. Quelques exemples (figure 3.5) ont été expérimentés en 1D sous Maple. Si leur complexité est faible grâce aux FFT, le problème reste le même que pour les filtres gaussiens : l'étendue de la "diffusion" est fixée, ce qui est mauvais lorsque deux contours sont proches.

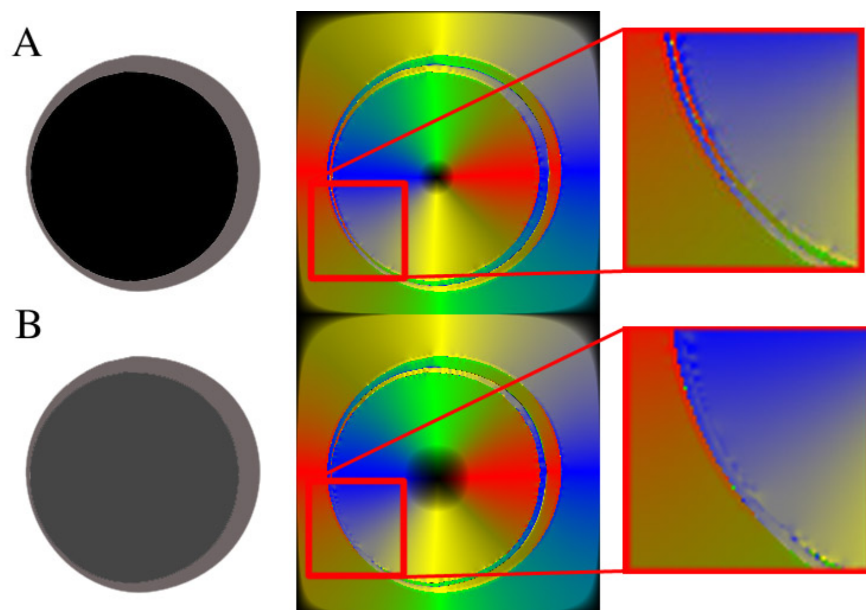


FIG. 3.4 – Apport du GVF généralisé : réduction de la dépendance force de diffusion / précision.
Les forts gradients (ou de force identique, cas A) restent bien distincts, quelle que soit leur distance. Par contre les détails de faible amplitude ne sont pas pris en compte : les faibles gradients trop proches d'un contour plus marqué sont supprimés (cas B, où le gradient dû au disque intérieur est plus faible que celui à l'extérieur).

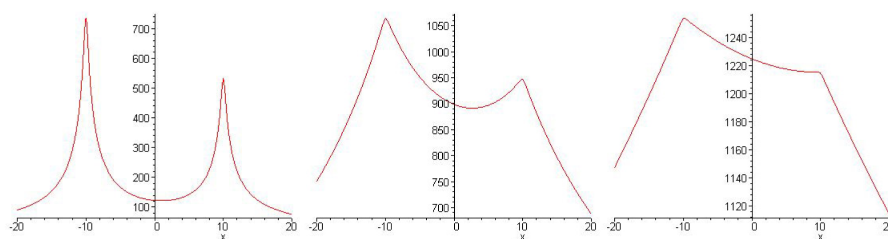


FIG. 3.5 – En une dimension, la diffusion des Diracs à partir de convolutions

Chapitre 4

Le multi-échelle

Les analyses multi-échelles ont permis d'introduire de nouvelles méthodes pour détecter les contours [17]. Si le GVF a étendu la capacité d'attraction du champ de forces, il reste à traiter la deuxième partie du problème, celui du choix des points d'attraction pour le contour actif.

Dans le chapitre précédent, le calcul du GVF s'appuie sur le champ de forces ∇f, où l'énergie f n'est que le module du gradient de l'intensité. Or cela ne permet pas de distinguer deux types de gradients, comme illustré pour un objet circulaire sur la figure 4.1. En effet, se superposent le gradient dû à une discontinuité - le contour - et le gradient qui dériverait d'une variation de l'intensité à la surface de l'objet. L'absence de distinction entre diffusion du contour et variation d'intensité au sein de l'image est source d'erreur dans la définition de la force d'attraction précédente.

Pour apporter une amélioration à ce problème, nous exposons trois possibilités : le détecteur multi-échelle comme énergie de base au calcul du GVF, un GVF multi-échelle ou l'algorithmique.

4.1 Détecteurs de contours

L'idée mise en oeuvre est d'appliquer à l'image initiale un détecteur de contour type détecteur de Canny (figure 4.2). Les points de contour sont définis comme des modules maximaux de la transformée en ondelette (dyadique)

$$Wf(u, 2^j) = I * \psi_{2^j}(u)$$

où

$$\psi = -\nabla G_\sigma \qquad \psi_{2^j} = 2^{-j}\psi(2^{-j}u)$$

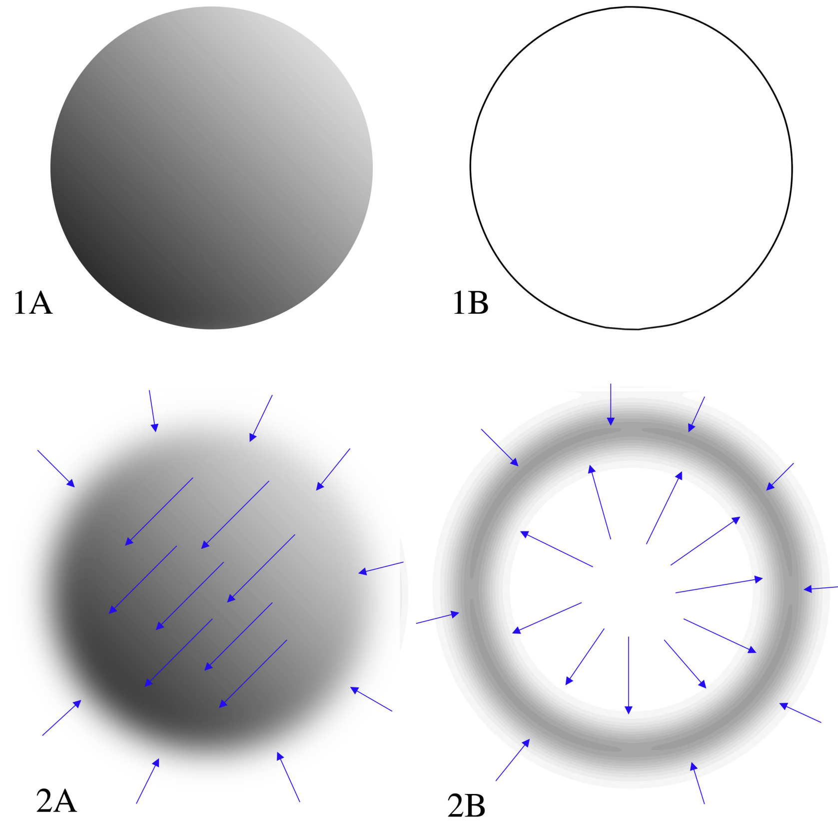


Fig. 4.1 – Deux types de gradients : sur l'image A, après diffusion (2A), se superposent le gradient initial, source d'instabilité, et le gradient de diffusion du contour (gradient d'attraction). Sur l'image B (image des discontinuités), le mauvais gradient est supprimé et seul le champ d'attraction aux contours est conservé

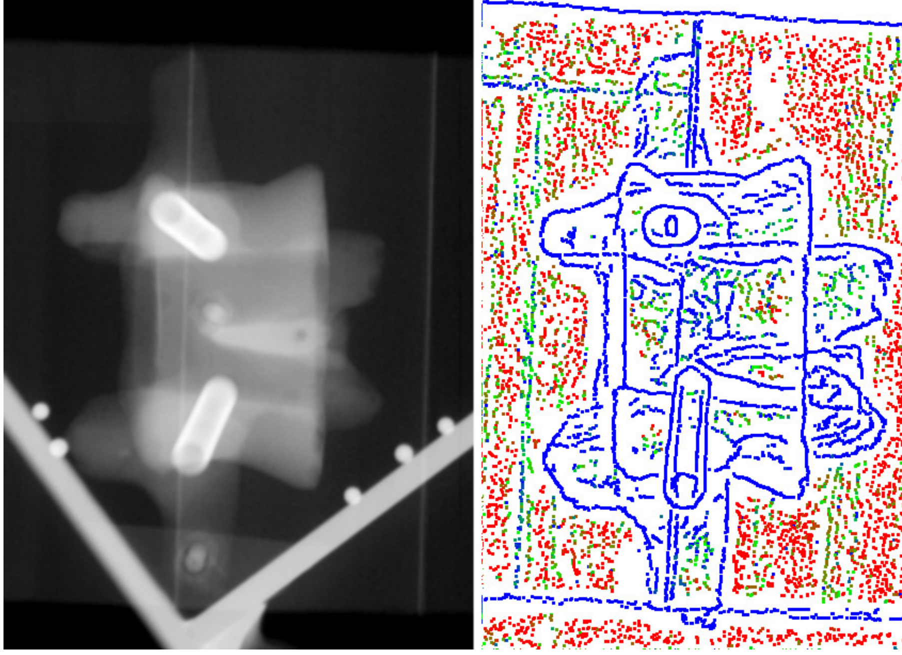


FIG. 4.2 – Image radio initiale et passage au détecteur de contours

Un point u est un module maximal si il est maximum local dans son voisinage suivant l'axe de direction $Wf(u,s)$.

Nous ne détaillerons pas le fonctionnement de ces algorithmes et utiliserons directement les résultats comme énergie f de départ au calcul du GVF.

4.2 GVF multi-échelle*

Il est possible d'étendre la généralisation du GVF pour gagner en performance. Lorsque l'on a introduit le GVF au chapitre précédent, on l'a appliqué au champ de force ∇f (où $f = |(\nabla G_\sigma) * I|$). On peut généraliser ce résultat en recherchant un nouveau champ vectoriel d'attraction $\mathbf{v}$ en minimisant la fonctionnelle

$$\int\int_{image} g(\mathbf{E})(u_x^2 + u_y^2 + v_x^2 + v_y^2) + h(\mathbf{E}) \sum_{j \leq n} |\mathbf{v} - \psi_{2^j} * f|^2 \ \ dxdy \qquad (4.1)$$

où $\mathbf{E} = \sum_j |Wf(u, 2^j)|$. L'équation d'évolution devient alors simplement

$$\nabla \cdot (g\nabla u) = h \sum (u - \psi^1_{2^j} * f) \qquad \frac{\partial u}{\partial n}_{|bords} = 0$$
$$\nabla \cdot (g\nabla v) = h \sum (v - \psi^2_{2^j} * f^i_y) \qquad \frac{\partial v}{\partial n}_{|bords} = 0$$

où les ψ^i sont les 2 composantes de ψ.

Interprétation E est d'autant plus fort que f est irrégulière sur un voisinage (proche d'un maximum de $Wf(u, s)$ à toutes les échelles). Si E est grand, le champ v va être le résultat de la minimisation de

$$\sum_{j \leq n} |\mathbf{v} - \psi_{2^j} * f|^2$$

Lorsqu'il y a discontinuité, on conserve ainsi et on moyenne l'information directionnelle de discontinuité sur toutes les échelles.

Lorsque f est régulière, le champ v minimise de préférence le terme $(u_x^2 + u_y^2 + v_x^2 + v_y^2)$, et est donc à variation faible : on diffuse les variations présentes aux limites.

En gros, ce doit être une sorte d'analogue en 2D de la recherche de suites de maxima locaux à travers les échelles. Un point de discontinuité (fort gradient) ne sera conservé que s'il correspond à un fort gradient aux échelles plus grandes, c'est-à-dire s'il est situé dans une zone correspondant à un contour. L'intérêt est de rendre le GVF moins sensible aux forts gradients isolés (bruit).

Remarques

- Il existe des cas particuliers où f est continue, alors qu'on est sur une suite de maxima à travers les échelles.
- Cela ne prend pas en compte la définition de contour selon Canny.
- On peut choisir $f = I$.
- On peut choisir $\psi = -(\frac{\partial^{(n)} G}{\partial x^{(n)}}, \frac{\partial^{(n)} G}{\partial y^{(n)}})$ pour éviter des instabilités en réduisant la force des gradients ne correspondant pas à des discontinuités.

Cette généralisation du GVF n'a pas été implémentée avant la rédaction du rapport.

4.3 Un algorithme*

L'idée est de diffuser l'attraction là où l'attraction est faible uniquement. On se donne pour cela plusieurs cartes d'attraction à différentes échelles, par exemple

un jeu de convolées de ∇f avec une gaussienne à différentes échelles.

Si la force d'attraction au contour est trop faible, l'algorithme utilise la carte d'attraction la plus grossière (gaussienne de forte variance). Si la force d'attraction est trop forte, il descend dans les échelles chercher l'attraction plus fine.

L'algorithme décrit ici est une meilleure version qui découle de celui implémenté (choisir le mode `m` du programme C). Ce dernier avait pour objectif de faire converger uniformément les points dans le champ d'attraction, en commençant par les points situés dans les zones de faible attraction (figure 4.3). Sa principale limitation était la présence de contours moins marqués dans les images.

4.3.1 L'algorithme

L'algorithme implémenté (en mode `m`, figure 4.3) consiste à

1. Définir un ensemble de points du contour actif où la force est plus faible que la moyenne
2. Éventuellement ne prendre qu'un sous-ensemble du précédent sous la forme d'une réunion d'intervalles
3. Déplacer ces points
4. Répéter les étapes 2 et 3 jusqu'à ce que le déplacement soit petit.

L'algorithme multi-échelles serait une amélioration qui consisterait à déplacer plutôt tous les points du contours en même temps, sous l'action d'une force modifiée, calculée de la façon suivante :

1. Etape 0 : La force initiale est donnée par une carte initiale J_0 (par exemple $J_0 = \nabla I$ [ou $J_0 = I * \psi_n$])
2. A l'étape k, si cette force en un point est trop faible, on lui ajoute la force J_{k+1} (où par exemple $J_{k+1} = J_k * G_\sigma$ ou $J_{k+1} = GVF(J_k)$ [ou resp. $J_k = I * \psi_{n-k}$]).

4.3.2 Apports

Portée d'attraction Si le contour actif est initialisé loin des forts gradients, il suffit de se donner un seuil suffisamment élevé pour ne pas tenir compte du bruit et aller chercher l'attraction due à la diffusion d'ordre supérieur (liée à la présence d'un contour). Le seuil est le seul paramètre à régler, son augmentation permet de sortir rapidement des minima locaux. Il est également possible d'implémenter le calcul d'un *seuil efficace* à partir de l'image initiale, avec quelques considérations statistiques de significativité [15].

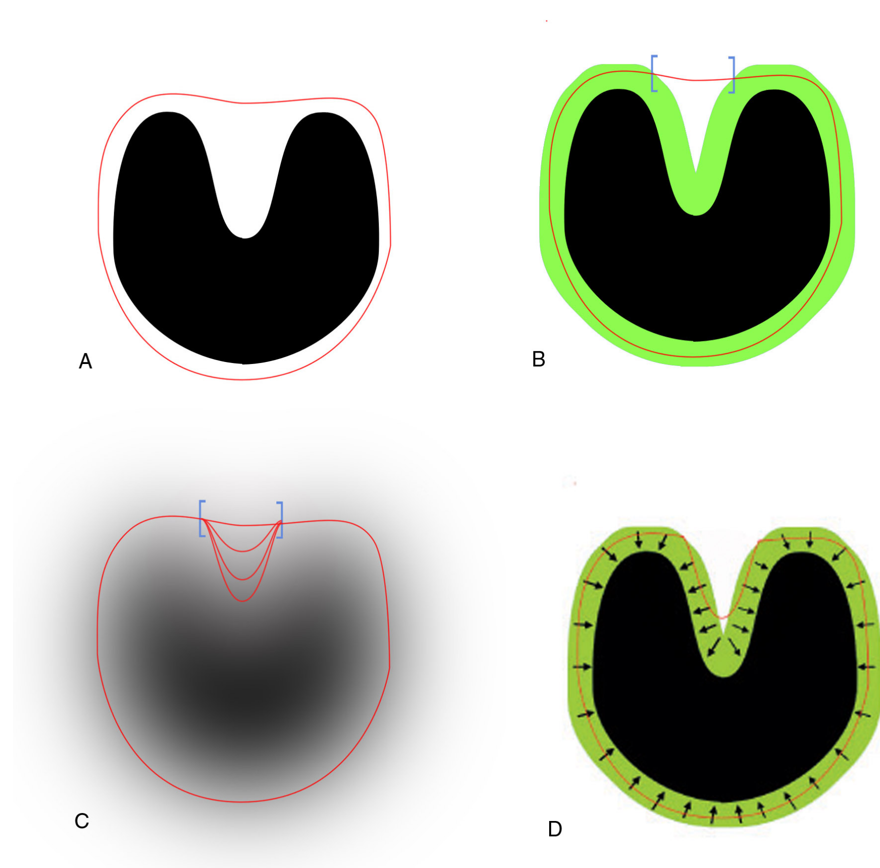


FIG. 4.3 – Description du mode m de l'algorithme C

Vitesse globale Cet algorithme permet de diminuer le nombre d'itérations nécessaires pour la convergence du contour actif et le calcul, dans la mesure où la force d'attraction est augmentée.

Le nombre d'itération devenant à peu près le même pour tous les points du contour en mouvement, le temps d'exécution de chaque itération dépend plutôt du nombre de convolutions nécessaires pour diffuser les gradients en chaque point : dans le cas $\theta = G_\sigma$, si d est la distance $||_2$ entre le contour réel et le contour initial, le nombre de convolutions gaussiennes est de l'ordre de $\sim log_2(d/\sigma)$. Ainsi, le temps d'exécution, qui était fixe dans le calcul du GVF, est lié à la précision de l'initialisation, soit dans notre exemple à la dextérité du chirurgien. Remarquons qu'en réalité il n'est pas nécessaire de calculer l'ensemble des cartes J_k, qu'il suffit de calculer les attractions localement au contour actif.

Concavités et détails Cela remplace le calcul GVF.

Absence de perte de précision Comme pour le prétraitement GVF, l'algorithme ne perd pas de précision où les gradients sont élevés [resp. les discontinuités fortes] (ie les contours).

4.3.3 Les limitations

La figure 4.4 illustre le cas où deux contours de nature similaire sont proches, rendant le problème très sensible à l'initialisation. On pourrait peut-être sortir de ce minimum local en s'autorisant une augmentation du paramètre α - mais cela n'a pas de sens car α est un paramètre intrinsèque à notre vision, assurant la continuité face au bruit. Il est dangereux de l'utiliser pour résoudre ce cas particulier. Le chapitre 5 étudie cette difficulté.

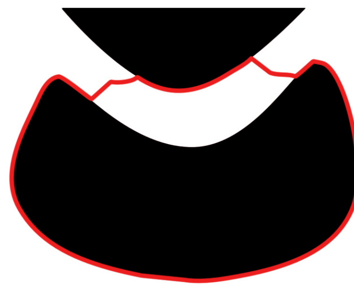


FIG. 4.4 – Le problème de contours multiples. Piégeage dans un minimum local de 2nd type

Chapitre 5

Force locale*

Dans l'espoir de pouvoir traiter le cas illustré figure 4.4, ce chapitre propose une observation de l'image du point de vue du contour actif, c'est-à-dire une description des propriétés de la force qui s'exerce le long du contour. La compréhension de l'évolution du contour actif n'est en réalité qu'un problème 1D.

5.1 Les minima locaux

5.1.1 Point de vue 2D

Précédemment a été introduite la notion de minimum local, c'est-à-dire une position stable du contour actif, qui ne correspond pas à un minimum visuellement évident.

On peut caractériser deux types de minima locaux :

1. Ceux correspondant à un contour secondaire, autour de l'objet à détourer par exemple
2. Ceux traversant des contours différents (figure 4.4)

Pour sortir d'une convergence vers un minimum local du premier type, il suffira souvent de diffuser plus ou d'élever le seuil de l'algorithme multi-échelles, la diffusion donnera raison au contour le plus marqué. Ici nous nous concentrons donc sur les problèmes du second type.

5.1.2 Point de vue 1D du contour

La représentation des forces d'attraction en tant que fonction du paramètre position sur le contour actif aide à la compréhension de son interaction avec l'environnement, responsable de son déplacement.

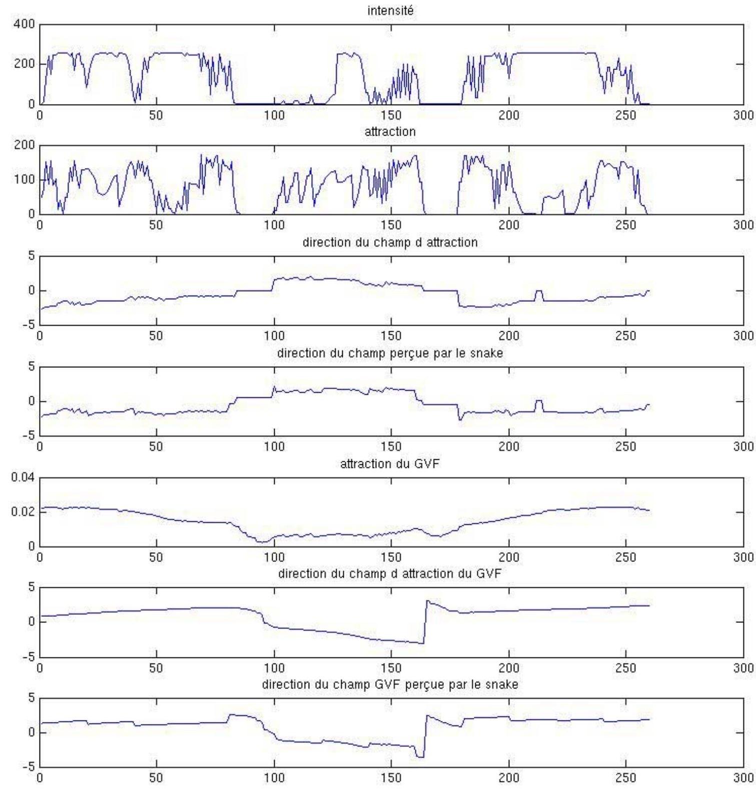


Fig. 5.1 – Représentation de la force d'attraction (intensité et direction) le long du contour initialisé dans la figure 5.2.

La figure 5.1 confirme le sentiment que la position locale du snake par rapport au contour se traduit dans l'intensité et la direction de la force d'attraction.

Inversion de l'attraction En manipulant le programme Matlab sur plusieurs images simples, on observe deux cas caractéristiques d'inversions du champ d'attraction

1. Le contour actif traversant un contour sur l'image
2. Une transition d'un contour à un autre, tel que sur la figure 4.4

Le deuxième cas se caractérise en outre par la colinéarité du contour et de la force locale au moment du "saut de contours". Or rappelons qu'un bon contour, solution au détourage d'un objet, satisfait la propriété suivante : la force est partout localement orthogonale au contour.

Reconnaissance des "sauts de contours" Considérant le premier cas d'inversion comme un cas peu probable de minimum local après quelques itérations de convergence du contour actif, il est donc possible de caractériser les sauts de contours type 4.4 par les 3 propriétés suivantes :

- Une inversion du champ d'attraction
- L'intensité de l'attraction proportionnelle à la distance au point d'inversion
- Des forces d'attraction colinéaires au contour

Un algorithme de propagation L'idée est de partir de ces points de "sauts de contour" pour sortir du minimum local et de propager l'adhérence à un seul contour sur les deux, une méthode qui n'a pas été implémentée avant la rédaction de ce rapport. Pour lancer cette propagation, il est par exemple possible d'inverser l'attraction sur quelques points après le point d'inversion, déplacer les points du contour en fonction de cette nouvelle force, recalculer le nouveau point d'inversion et répéter ces étapes tant qu'il y a inversion.

5.2 Force orthogonale - Une solution à la reparamétrisation

La reparamétrisation Dans le cas général, afin de conserver un bon échantillonnage du contour, il est nécessaire de reparamétrer le contour actif au cours de son évolution. Cela vient en partie des faibles rétrécissements ou allongements (par exemple l'initialisation se fait souvent à l'extérieur du contour, d'où une contraction) mais surtout des forces s'exerçant sur le contour actif. Par exemple, dans les illustrations 2.3 et 4.4, le contour actif "décolle" du contour en suivant les lignes

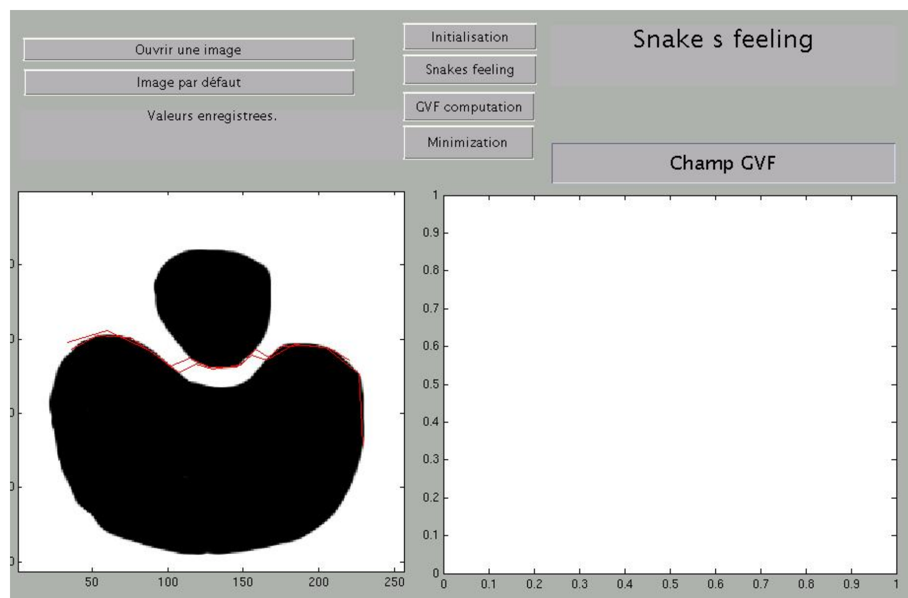


FIG. 5.2 – Sauts de contour : inversion d'attraction et décroissance de l'attraction au point de sauts de contours sont représentés sur la figure précédente 5.1

de champ du champ d'attraction ; le champ d'attraction va donc ou bien dilater ou bien étirer les points du contour.

La reparamétrisation étant coûteuse à la fois en calcul et en précision, il est meilleur de s'en passer, comme proposé ici.

Force orthogonale La composante tangentielle des forces agissant sur le contour actif n'a qu'un rôle parasite, qui n'a pour effet que d'étirer les points. Il est possible de la supprimer en calculant localement la composante normale de cette force par rapport au contour.

La conséquence est une légère rigidification du contour qu'il est possible de compenser par une diminution des paramètres d'élasticité.

Implémentation On peut exprimer les coordonnées discrètes du vecteur tangent à partir de 4 points successifs du contour

$$\alpha = 1/3\left((x_{i+1} - x_{i-1}) + 1/2(x_{i+2} - x_{i-2})\right)$$

$$\beta = 1/3\left((y_{i+1} - y_{i-1}) + 1/2(y_{i+2} - y_{i-2})\right)$$

et en déduire la force orthogonale donnée par

$$(f_\perp)_x = -\beta \frac{-\beta f_x + \alpha f_y}{\sqrt{\alpha^2 + \beta^2}}$$

$$(f_\perp)_y = \alpha \frac{-\beta f_x + \alpha f_y}{\sqrt{\alpha^2 + \beta^2}}$$

5.3 Force globale

De façon à recaler rapidement la forme initiale sur le contour réel, le mode `global` de l'implémentation C calcule la force qui s'applique au système {*contour actif*} par sommation des forces locales s'exerçant sur le contour et déduit la direction de la translation globale à effectuer.

Intérêt Rapidité et stabilité sont les atouts de ce mode de convergence. Les défauts de l'image (superpositions d'objets, bruits...) ne provoquent pas de perturbations. La forme initiale n'étant pas déformée, le mode global s'avère un outil performant pour caler un modèle statistique initial de vertèbres sur une vertèbre de la radio.

Remarque L'implémentation d'un moment global qui permettrait d'effectuer une rotation globale de la forme initiale n'a pas abouti à des résultats stables. Le moment global (des forces ou des forces orthogonales au contour) n'a pas une signification claire. De plus, dans le cas de formes plus ou moins circulaires, l'instabilité devient évidente.

Chapitre 6

Application aux images médicales

Dans ce chapitre sont présentés les résultats de la méthode des contours actifs sur des images radio réelles, composées d'os, de tissus et d'instruments se superposant.

6.1 Résultats

La radiographie a pour taille environ 400×600. Le calcul du GVF s'effectue en une trentaine de secondes (avec un code plus ou moins optimisé). La convergence du contour actif est rapide et efficace : 5 secondes suffisent à corriger une initialisation correcte (30 pixels de précision), ce qui est négligeable devant l'initialisation manuelle opérée par le chirurgien.

Les limitations à prendre en compte On retiendra des expériences que

- La méthode brute est relativement instable lorsque le contour est mal initialisé ou simplement translaté (figure 6.3). Elle marche très bien dans le cadre de faibles déformations.
- Le GVF aide pour beaucoup dans la tolérance aux conditions initiales. Son défaut est sa complexité.
- Le mode global (translation) est très efficace pour recaler linéairement le contour initial (figure 6.3), par contre il ne prend pas en compte les déformations.

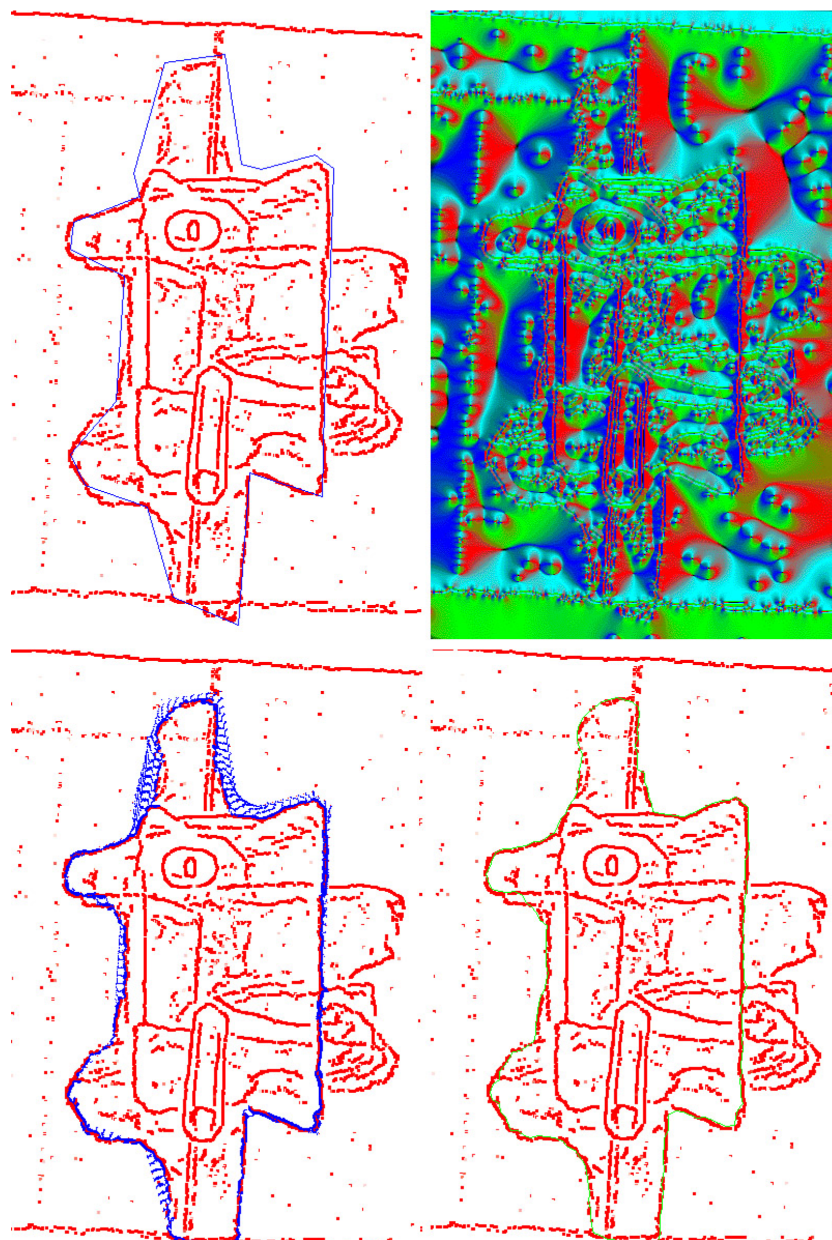


FIG. 6.1 – La détection de la vertèbre.
Dans l'ordre, initialisation du contour, GVF appliqué à 4.2 seuillé, convergence, contour final (vert).
Aucun défaut de convergence, pour une élasticité de 10^{-6}

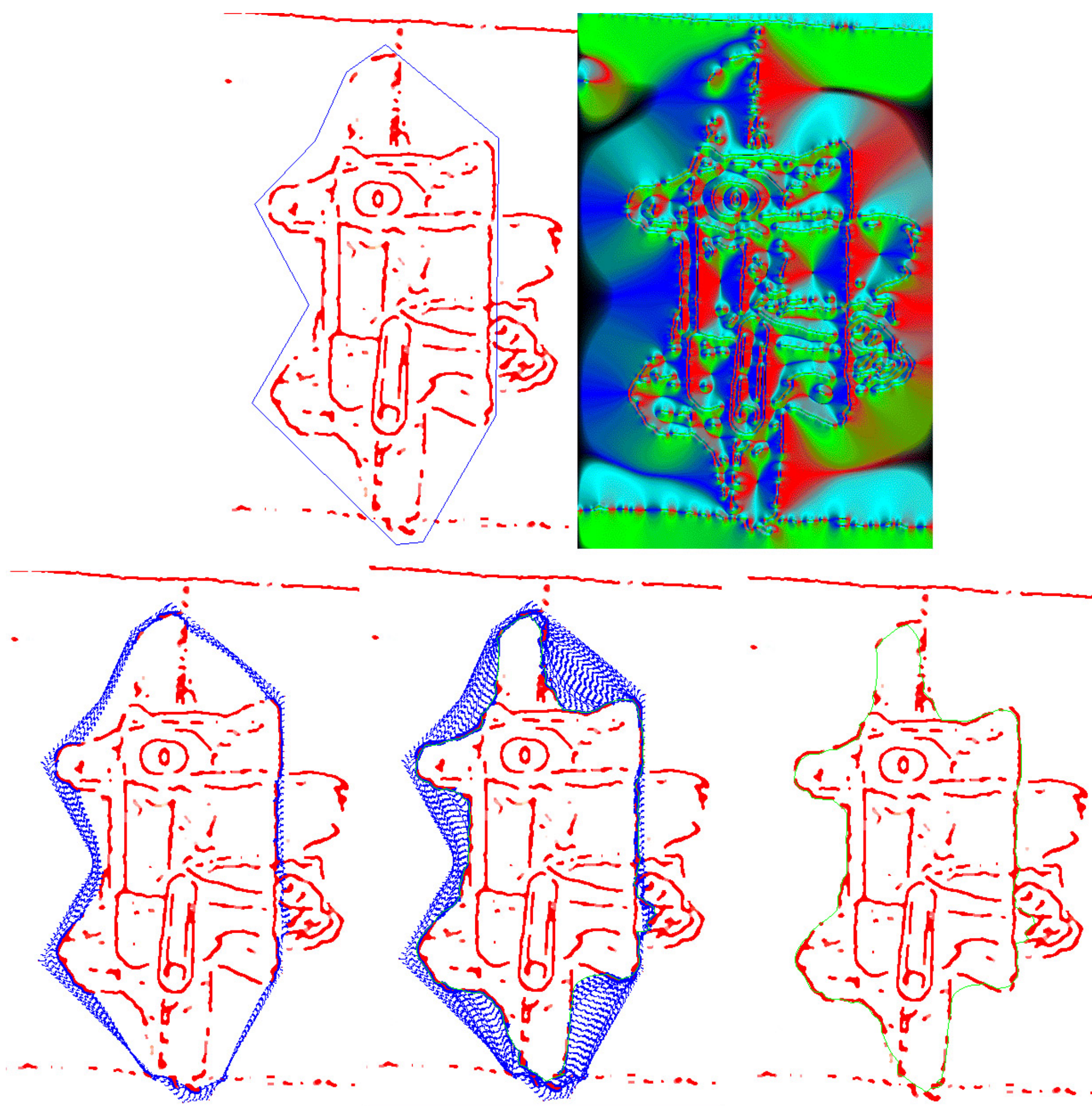


Fig. 6.2 – La détection de la vertèbre pour une initialisation approximative. Un seuillage et un filtre de débruitage ont été appliqués à l'image 4.2 obtenue par le détecteur de contours.
Un défaut de convergence subsiste dans cet exemple (même élasticité que précédemment), une meilleure initialisation l'aurait compensé.

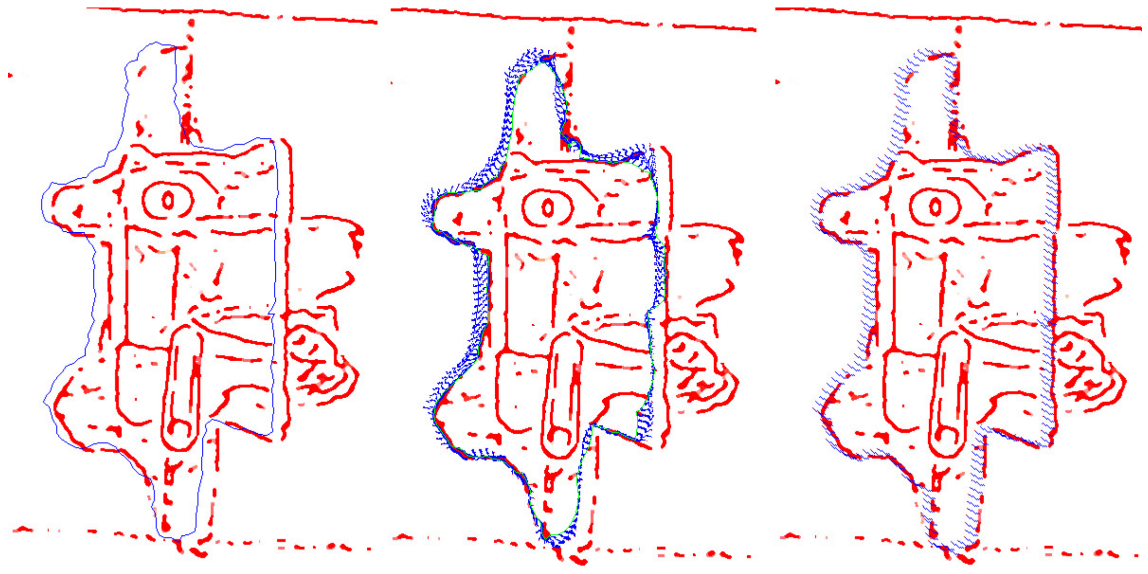


Fig. 6.3 – La détection de la vertèbre pour une bonne initialisation mais translatée.
Dans l'ordre, le contour initial, la convergence par les contours actifs, la convergence en mode global (translation).
Le mode global est bien meilleur pour faire un "recalage linéaire".

6.2 Une solution rapide et efficace

Sachant les atouts et les défauts de chaque traitement on peut organiser l'acte suivant le plan suivant :

1. Calcul de la forme de la vertèbre du patient, étape 1 (lent) : sur les radiographies prises sous deux angles au moins, on applique les différents filtres et le GVF. Le chirurgien (ou l'ordinateur) choisit une forme prédéfinie de vertèbre issue de modèles statistiques. Il la positionne, modifie certains points éventuellement.
2. Calcul de la forme, étape 2 (lent) : les contours actifs vont permettre de modifier ces projections et les coller au plus près des vertèbres. Les données seront traitées par le logiciel de reconstruction 3D qui affine ses critères dans le choix du modèle de vertèbre. Par une série d'aller-retour entre la base de donnée et les contours actifs, l'ordinateur peut déterminer la forme de la vertèbre 3D.
3. Début de l'opération : initialisation, recalage linéaire des formes sur les contours (avec éventuelle aide du chirurgien)
4. Au cours de l'opération (temps réel) : "tracking" de la vertèbre avec alternance recalage linéaire/contours actifs

Une part importante du travail sera d'adapter les différents paramètres à partir d'une base de données de radiographies et de s'assurer du fonctionnement sous tous les cas de figure.

6.3 Les axes de recherche

La méthode des contours actifs a connu d'autres développements qui semblaient moins adaptés à notre problème : force de ballon [4], méthode par level-set...

A noter, en conclusion, qu'un gros défaut de la méthode reste l'absence de contraintes de forme, outre la régularité du contour : seule l'initialisation permet d'induire une "idée" de la forme au contour actif. Il est certes possible d'introduire de nouvelles formes d'énergies, mais les développements existants (comme [13]) rendent le problème difficile à minimiser.

Chapitre 7

Conclusion personnelle sur le stage

Je tiens à remercier Valérie Perrier pour l'accueil qu'elle a bien voulu me réserver, le thésard Olivier Le Cadet et les élèves de l'ENSIMAG Anne-Thérèse Bourreau, Benjamin Roure, Sylvain Vallaghé et Sofiane Akkouche pour leur aide dans la réalisation de ce projet.

Ce stage, qui s'inscrit dans le prolongement des majeures de Mathématiques appliquées et d'Informatique de l'X, m'aura notamment permis de mettre en perspective le cours d'optimisation numérique de Grégoire Allaire, le cours de vision et d'analyse des images digitales de Jean-Michel Morel et les cours d'informatique. C'est aussi et surtout dans le domaine de l'informatique appliquée aux images que les efforts se sont concentrés au long de ce stage ; l'enseignement à l'X se basant de préférence sur le langage Java, il a fallu faire preuve d'une nécessaire adaptation vers le langage C.

Il est certain que l'expérience acquise en programmation dans le cadre des images digitales s'avérera une bonne préparation au DEA Mathématiques Vision & Apprentissage de l'ENS Cachan.

Chapitre 8

Annexe

8.1 Le programme

8.1.1 Utilisation

Quelques touches permettent de contrôler le déroulement du programme de l'interface :

i initialise le snake (lorsque les points d'initialisation du contour ont été sélectionnés, avant de lancer l'algorithme)

r lance l'algorithme de contour actif

q ou ESC quitte l'algorithme (si la norme de la différence ne converge pas en dessous du seuil fixé) ou quitter le programme si l'algorithme est terminé.

INS réinitialise le tout

D'autres permettent de choisir les paramètres :

p passe en mode "pas à pas". Une fois le contour initialisé avec 'i', il faut appuyer sur la touche 'r' pour passer d'une itération algorithmique à l'autre.

s sauve la sélection de points (initialisation) ou le contour final (fin d'algorithme)

l charge le contour initial (enregistré dans le fichier selection_to_load.txt)

m mode multi-scale (progression parallèle)

o mode "force orthogonale au contour"

g mode force globale (déplacement du contour actif sans modification de la forme initiale)

8.1.2 Une brève description de l'implémentation C

Tous les fichiers sources peuvent être téléchargés à l'adresse
`http://christopher.bourez.free.fr/contours_actifs/`
Ces programmes écrits en C marchent sous Unix ou Windows. Pour les compiler sous Unix, utiliser la commande `make` (associé aux fichiers `makefile`). Sous Windows, les espaces de travail (*workspace*) sont définis pour Visual C++ 6.0 dont la licence est gratuite pour un usage non commercial.

Les librairies La compilation de ces programmes requiert l'installation de deux librairies

-libtiff La librairie libtiff est très utile pour lire et écrire des fichiers tiff.

-GLUT La librairie GLUT est une librairie pour OpenGL (GL utility toolkit) qui permet la création de fenêtres et offre la portabilité. Elle permet de gèrer non seulement le graphisme de façon conviviale mais également la souris et le clavier.

Les espaces de travails (*workspaces*)

/src/ Il contient deux programmes principaux. `gvf.c` permet de calculer le GVF généralisé sur l'image `in.tif` et l'enregistre les valeurs dans deux fichiers textes `grad_x.txt` et `grad_y.txt` ainsi qu'une image représentative `gvf.tif`. L'autre programme `snake.c` utilise ces sorties, initialise le contour sous la forme d'un cercle et itère suivant la méthode des contours actifs. Il enregistre le résultat dans `out.tif`.

/interface/ A partir des fichiers `in.tif`, `gvf.tif`, `grad_x.txt` et `grad_y.txt`, il donne une interface conviviale de `snake.c` permettant de sélectionner les points d'intialisation du contour et de visualiser la progression.

Les méthodes Afin de rendre la lecture des fichiers `.cpp` plus facile, nous donnons ici la signification des différentes variables utilisées et l'action des différentes méthodes.

Le squelette

- Les variables de calcul matriciel : `double *A, L, U ;`
- l'élasticité et la rigidité du contour (tableaux) : `double *alpha, beta ;`
- les paramètres pour l'itération : le nombre de points NB_POINTS, les pas TAU et H, le seuil pour l'arrêt de l'itération EPSILON, le nombre d'itérations maximales NB_ITER_MAX, des constantes pour l'élasticité et la rigidité ALPHA BETA

- la force le long du contour : `double *Fx, *Fy;`
- les points du contour actif :
  `double* Xold, *Yold, *Xnew, *Ynew, *Xaux, *Yaux;`
- les points du contours non fixés : `int* Mobiles ;`
- le champ d'attraction du snake : `double* gradx ,grady ;`
- les paramètres de l'image : `int width, height;`
- les images de travail (entrée, gvf, affichage gl) :
  `unsigned char* imagein, imagedep, imagegvf, imagegl;`
- les tableaux de points de l'initialisation du contour : `int* posx, posy; int nbpointdessin;`
- la gestion des phases du programme (initialisation, int run = 0 ; int initialized = 0 ; int launchinitialization = 0 ; int view = 0 ; int nbview = 2 ; int fini = 0 ;
- les parametres booléens :
  `int ortho = 0;` pour le choix de la force entière/orthogonale au contour ;
  `int multi = 0;` pour le choix du mode multi-échelles ;
  `int pasparpas = 0;` pour le choix du mode pas par pas ;
  `int global = 0;` pour le choix du mode de repositionnement global (calcul de la force le long du contour et translation sans modification de la forme) ;
  `int rotate = 0;` pour basculer entre translation et rotation/homothétie à appliquer aux points du contour initial.

Les méthodes

- *definitionA, Decomposition LU, Inversion* servent au calcul des DF
- *normeVectorielle* calcule la norme du mouvement du contour entre deux itérations
- *load_bmp, create_bmp* gère les images BMP
- *homothetie, rotation, move (translation)* permettent d'ajuster le contour initial
- *load_Selection, enreg_Selection* permettent de charger/enregistrer le contour initial et final
- *initializingsnake* initialise le contour actif
- *Stockage_gradients, recup_gradient* stocke ou récupère les champs de forces du GVF dans des fichiers textes
- *visual_couleur* donne une représentation des champs vectoriels (norme = intensité et direction = couleur)
- *Draw, drawfinal* gèrent le dessin
- *MouseSelect, GestionClavier, GestionSpec* gère la souris, le clavier, les touches spéciales
- *display*, fonction de boucle d'affichage/calcul
- *afficheContour* affiche les contours intermédiaires
- *Fext, FextOrtho* calcule la force le long du contour

- *reparam* reparamétrise le contour actif au cours des itérations

Le reste sert au débogage et à l'affichage intermédiaire.

Bibliographie

[1] *Snakes : Active Contour Models* Michael Kass, Andrew Witkin and Demtri Terzopoulos

[2] *Image understanding : Active contour models - snake* (Chap 8, Part II, Digital Image Processing)

[3] *On Active Contour Models and Balloons*, Laurent Cohen

[4] *Modèles déformables*, Laurent Cohen

[5] *Deformable Models and Minimal paths for image analysis*, Laurent Cohen

[6] *Multi-Resolution algorithms for Active Contour Models*, Bertrand Leroy, Isabelle L. Herlin and Laurent Cohen

[7] *A wavelet-frame based Image Force Model for Active Contouring algorithms*, Hsien-Hsun Wu, Jyh-Charn Liu, and Charles Chui

[8] *Snakes, Shapes, and Gradient Vector Flow*, Chenyang Xu and Jerry L. Prince

[9] *Gradient Vector Flow : a new external force for Snakes*, Chenyang Xu and Jerry L. Prince

[10] *Deformable Models with application to human cerebral cortex reconstruction from magnetic resonance images*, Chenyang Xu

[11] *Global Optimality of Gradient Vector Flow*, Chenyang Xu et Jerry L. Prince

[12] *Wavelet-based Active Contour Model for Object Segmentation and Tracking in Video Sequences*, National Taiwan University, Institute of Information Science, Institute of Microbiology and Immunology, Taiwan

[13] *Etude d'une nouvelle classe de contours actifs pour la détection de routes dans les images de télédétection*, Marie Rochery, Ian Jermin, Josiane Zerubia, Projet ARIANA, Sophia Antipolis

[14] *Analyse numérique et optimisation*, Grégoire Allaire, cours de l'école Polytechnique.

[15] *Bases mathématiques de l'analyse des images digitales*, Jean-Michel Morel, cours de l'école Polytechnique

[16] *Cours du traitement du signal et de l'image* , Jean-Michel Morel, Ecole Normale Supérieure de Cachan

[17] *Une exploration des signaux en ondelettes*, Stéphane Mallat, édition de l'Ecole Polytechnique